THM - BountyHunter Writeup

Zackary Jordan

01/25/2026

1 Target Information

- **Target IP Address:** 10.67.137.128

2 Reconnaissance

I used RustScan to identify open ports on the target system and piped it into nmmap scripts.

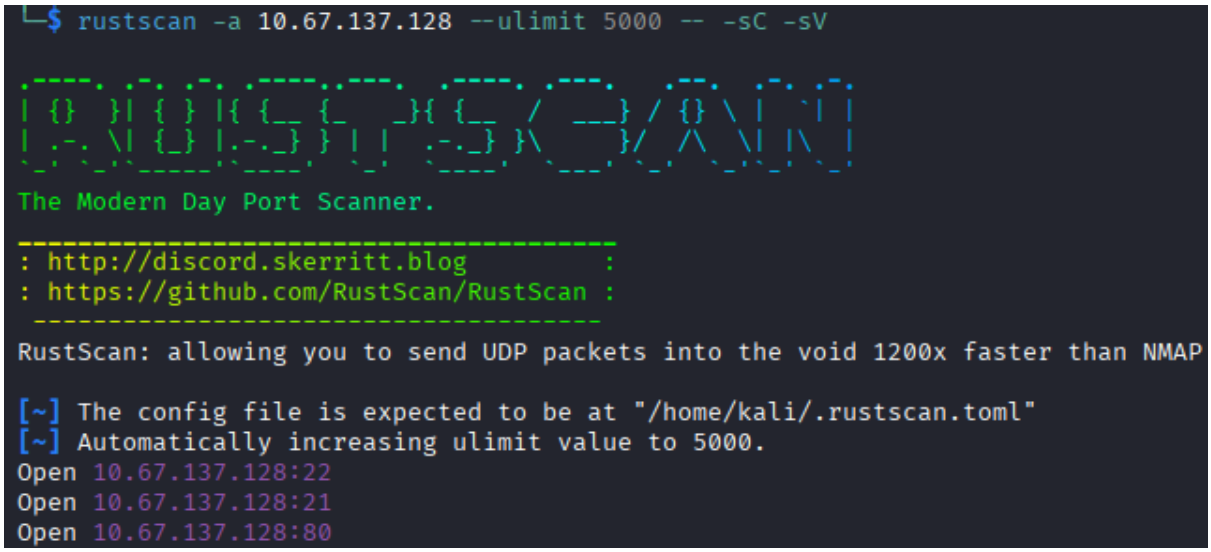

Figure 1: RustScan showing open ports on the target

3 Enumeration

3.1 Nmap Enumeration

I then used Nmap scan with service dependent scripts to gain more information about the target.

```
nmap -sV -sC 10.67.137.128
```

```
PORT   STATE SERVICE REASON         VERSION
21/tcp open  ftp     syn-ack ttl 62 vsftpd 3.0.5
| ftp-anon: Anonymous FTP login allowed (FTP code 230)
|_Can't get directory listing: PASV failed: 550 Permission denied.
| ftp-syst:
|   STAT:
| FTP server status:
|      Connected to ::ffff:192.168.162.149
|      Logged in as ftp
|      TYPE: ASCII
|      No session bandwidth limit
|      Session timeout in seconds is 300
|      Control connection is plain text
|      Data connections will be plain text
|      At session startup, client count was 3
|      vsFTPd 3.0.5 - secure, fast, stable
|_End of status
22/tcp open  ssh     syn-ack ttl 62 OpenSSH 8.2p1 Ubuntu 4ubuntu0.13 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   3072 69:cf:2c:73:7d:f6:89:5d:47:09:2b:1e:ec:a0:0c:50 (RSA)
| ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABgQDMcvb7Q/jteJ/GXqgZTTOWTmIpulTpzaFPff0K16xqnN+JGGKGo7vCWEGLh
Lc0hZB2UwM10uE5wNOqoLjY8Pd4HeMhHShD8KMNt4NEjJ/7EO/5R6f+zgMvDqsECVsEceT/j/pd2LDlyuWEi6Mh9xbxPBaIi4SL
|   256 41:4b:35:60:93:2c:35:5a:cf:35:cd:ba:fe:de:c9:59 (ECDSA)
| ecdsa-sha2-nistp256 AAAAE2VjZHNhLXNoYTItbmlzdHAyNTYAAAAIbmlzdHAyNTYAAABBBLnsM39Xt8rtBZI54tXc1SyU5
|   256 1f:19:8d:0d:8f:df:e8:20:c3:33:fc:4b:8a:84:b1:56 (ED25519)
|_ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAIBXraDgHpTwH5p84T9SXxeo701GJ+EoGAGtqT3qUljKj
80/tcp open  http    syn-ack ttl 62 Apache httpd 2.4.41 ((Ubuntu))
|_http-server-header: Apache/2.4.41 (Ubuntu)
| http-methods:
|_  Supported Methods: HEAD GET POST OPTIONS
|_http-title: Site doesn't have a title (text/html).
Service Info: OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel

NSE: Script Post-scanning.
NSE: Starting runlevel 1 (of 3) scan.
Initiating NSE at 22:12
Completed NSE at 22:12, 0.00s elapsed
NSE: Starting runlevel 2 (of 3) scan.
Initiating NSE at 22:12
Completed NSE at 22:12, 0.00s elapsed
NSE: Starting runlevel 3 (of 3) scan.
Initiating NSE at 22:12
Completed NSE at 22:12, 0.00s elapsed
Read data files from: /usr/share/nmap
Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 11.70 seconds
           Raw packets sent: 7 (284B) | Rcvd: 4 (172B)
```

Figure 2: Nmap scan results showing FTP, SSH, and HTTP services

3.2 FTP Enumeration

I first searched the FTP service and saw it allowed anonymous login. After doing so I found the directory revealed two files.

```
ftp> ls
550 Permission denied.
200 PORT command successful. Consider using PASV.
150 Here comes the directory listing.
-rw-rw-r--    1 ftp      ftp           418 Jun 07  2020 locks.txt
-rw-rw-r--    1 ftp      ftp            68 Jun 07  2020 task.txt
226 Directory send OK.
```

Figure 3: Anonymous FTP login revealing locks.txt and task.txt

I then ran both `get locks.txt` and `get task.txt` to download the files for future use. After which, I opened `task.txt` and saw the following message:

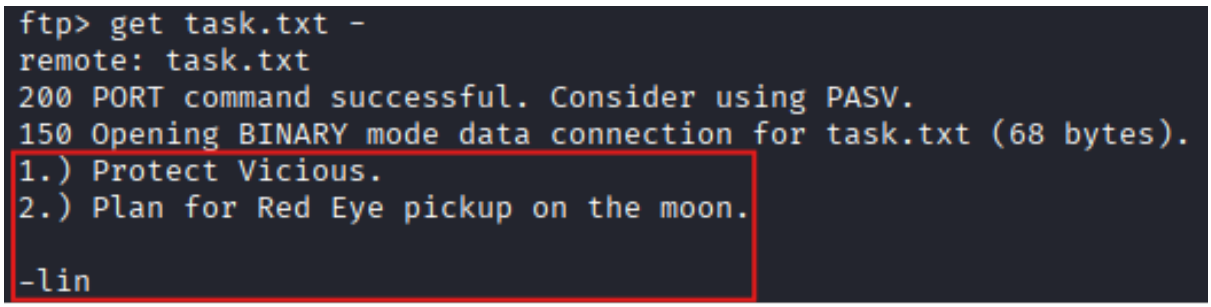

Figure 4: Contents of task.txt referencing user lin

3.3 Web Enumeration

After seeing all possible info from ftp I began http Enumeration. I accomplished this by using the FFUF command:

```
ffuf -w /usr/share/wordlists/dirb/big.txt -u http://10.67.137.128/FUZZ
    -mc 200,204,301,302,307,401,403 -s
```

Doing so, it revealed multiple web directories to investigate.

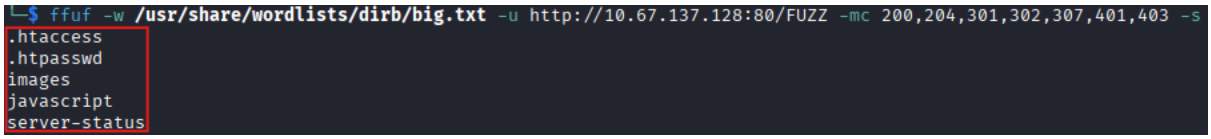

Figure 5: FFUF discovering common web directories

After checking all of them and inspecting the pages, I found no useful information.

4 Exploitation

With no other services to try, I attempted to SSH brute-force based on information gathered, using the username `lin` and passwords from `locks.txt` uisng the hydra command:

```
hydra -l lin -P locks.txt ssh://10.67.137.128
```

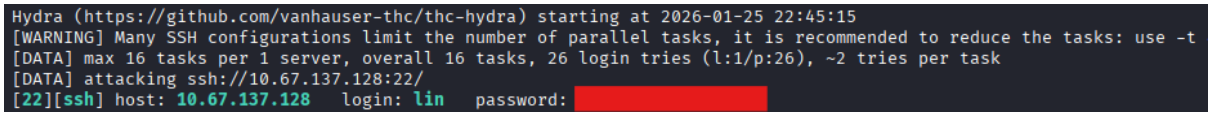

Figure 6: Hydra successfully identifying valid SSH credentials

With this success, I used the recovered credentials to login as `lin`.

5 Post-Exploitation

After gaining initial access, I did basic enumeration commands `whoami, id, uname -a` for proper identification. After which, I found the user flag was located in the home directory of `lin`.

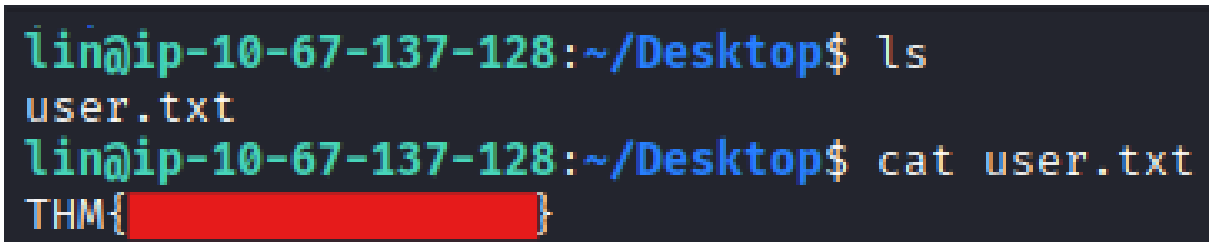

Figure 7: User flag retrieved from lin's home directory

From here, I did a search for SUID binaries which revealed: `/bin/tar`.

```
find / -type f -perm /4000 2>/dev/null
```

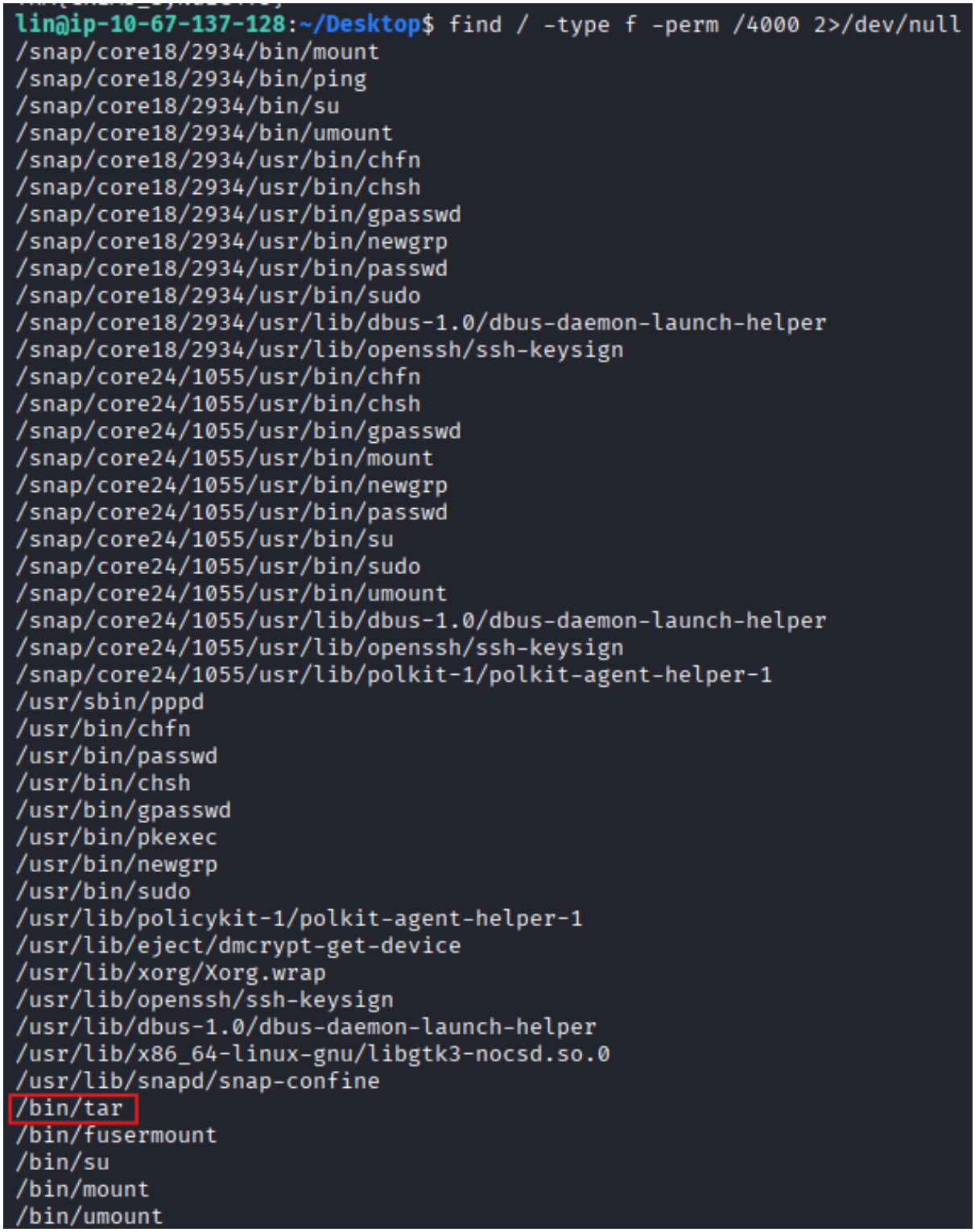

Figure 8: SUID binary /bin/tar discovered

6 Privilege Escalation

Using GTFOBins, I found a possible privilege escalation technique abusing the SUID `tar`.

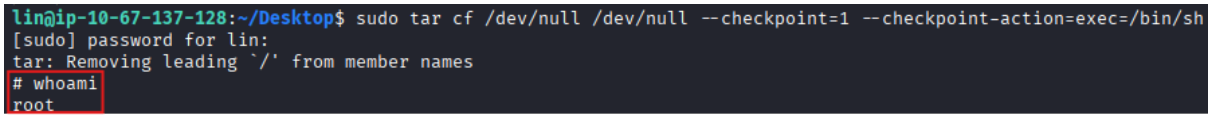

Figure 9: Privilege escalation via SUID tar resulting in root shell

After the command worked, I confirmed I had Root access and then retrieved the root flag.

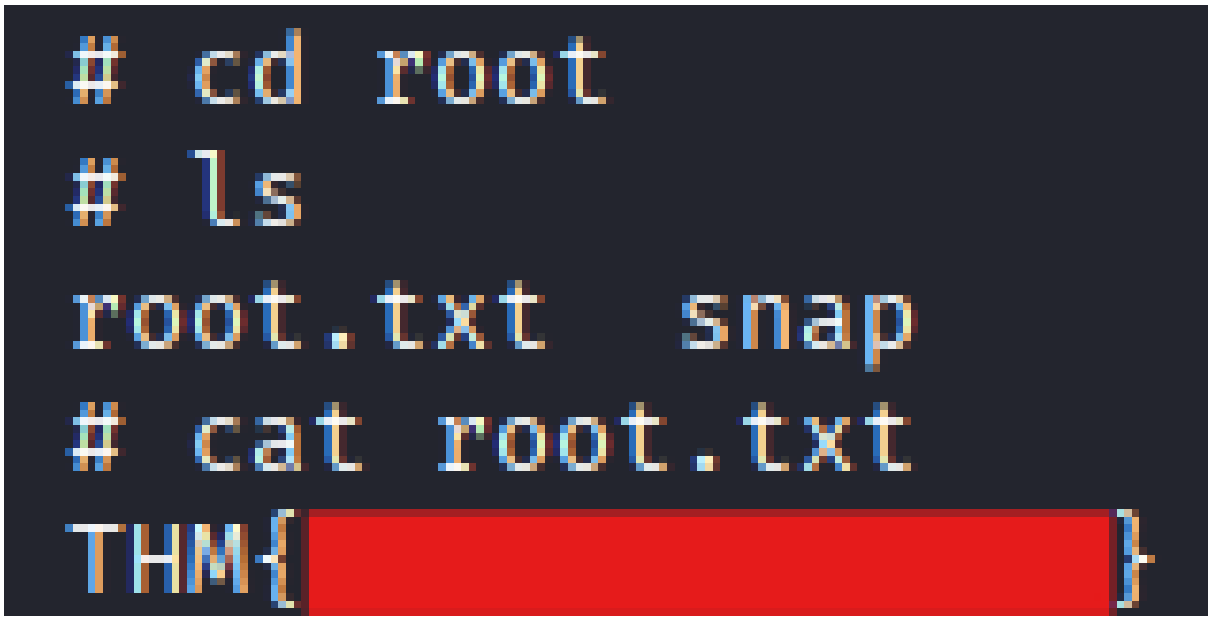

Figure 10: Root flag successfully retrieved

7 Conclusion

This is my writeup for THM-BountyHunter, sorry for the 8bit screenshots
– Zackary Jordan